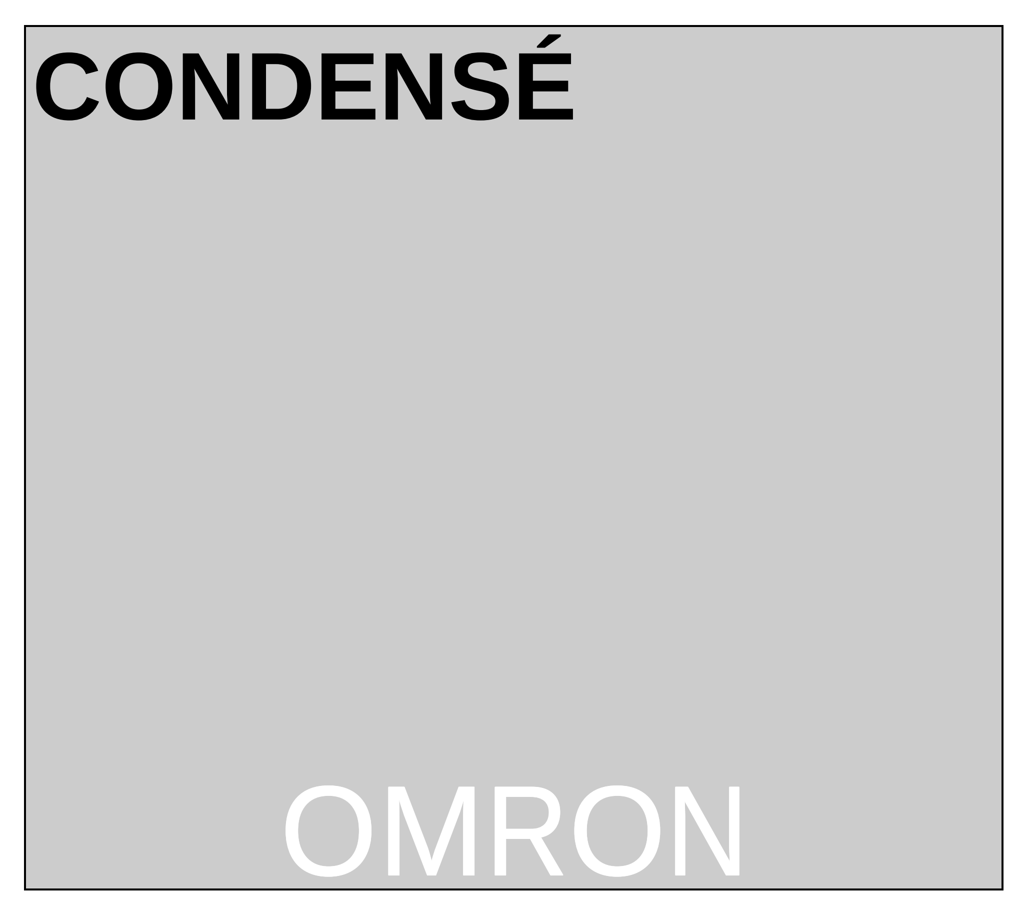
Protocol Macro

Cx-Protocol

Logiciel d'élaboration de protocole



Avertissement

Cette documentation est destinée à faciliter la mise en œuvre du matériel omron. Certains détails sont volontairement occultés pour ne pas provoquer de confusion. Malgré tout le soin apporté à la réalisation de cette documentation, omron ne pourra être tenu pour responsable des erreurs ou omissions et de leur conséquences. Cette documentation pourra être modifiée sans préavis et ne présente aucun engagement de la part d'omron.

Sommaire

1 CX-PROTOCOL V1.2 4

1.1 CARACTÈRISTIQUES 4
1.2 RACCORDEMENT RS232C 4

2 CONFIGURATION 5

2.1 AUTOMATE CS1 5
2.2 AUTOMATE CJ1 5
2.3 AUTOMATE C200HX/HG/HE 6
2.4 AUTOMATE CQM1H 6

3 PRINCIPE DE PROTOCOL MACRO 7

L'INSTRUCTION PMCR SUR CS1/CJ1 7
3.2 L'INSTRUCTION PMCR SUR C200HX/HG/HE ET CQM1H 7

4 ELABORATION D'UN PROTOCOLE 8

4.1 CRÉATION D'UN PROTOCOLE 8
4.2 SÉLECTION DE LA CARTE DE COMMUNICATION PMSU 9
4.3 ELABORATION DES SÉQUENCES 9
4.3.1 Création d'une séquence 9
4.3.2 Méthode de contrôle des signaux de transmissions 10
4.3.3 Création d'une étape 11
4.4 ELABORATION DES MESSAGES 12
4.4.1 Création d'un message 12
4.4.2 Détermination des délimiteurs de trame 13
4.4.3 Détermination de la méthode de contrôle d'erreur 13
4.4.4 Détermination du champs longueur [Length] 13
4.4.5 Détermination de l'adresse du destinataire [Address] 14
4.4.6 Détermination de la trame [Data] 14
4.4.7 L'adressage relatif 15
4.5 EXEMPLE 1 17
4.6 EXEMPLE 2 18
4.7 MATRICE 19
4.7.1 Principe 19
4.7.2 Création d'une matrice 20

5 MISE EN ŒUVRE DU PROTOCOLE 21

5.1 CONFIGURATION DE LA CARTE PMSU 21
5.2 TRANSFERT DU PROTOCOLE MACRO 22
5.3 LA FONCTION TRACE 22
5.4 DRAPEAUX SYSTÈME 23

1Cx-Protocol V1.2

1.1 Caractèristiques

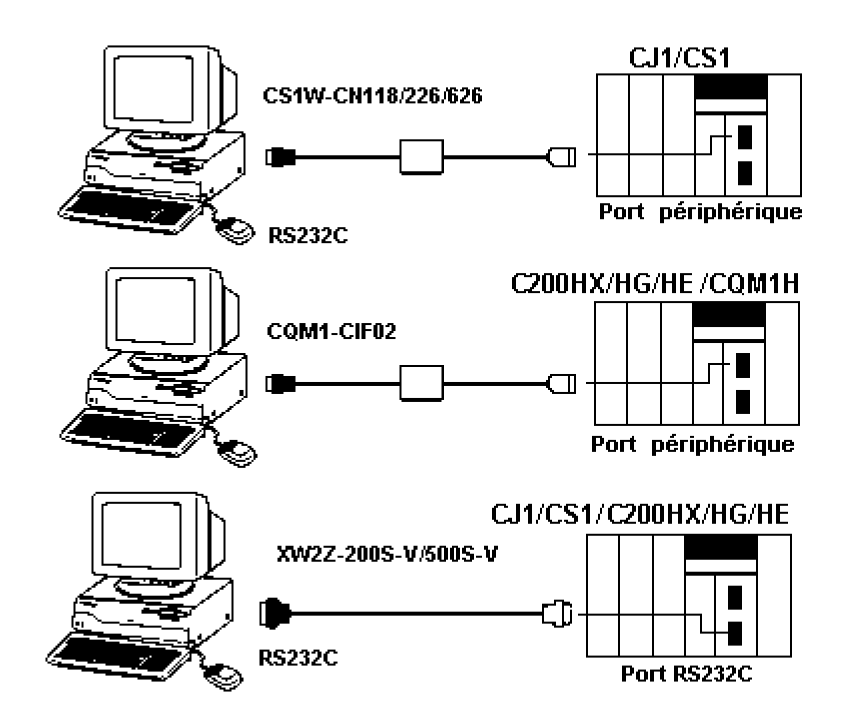
Cx-Protocol est un logiciel destiné à l'élaboration de protocole d'envoi et réception de donnée à partir des cartes PMSU (Protocol Macro Support Unit) et via RS232C, RS422 et RS485. Un protocole macro est constitué de séquences comportant des messages destinés à être envoyés ou bien reçus. Le protocole macro est élaboré puis transféré dans la carte PMSU à l'aide du logiciel Cx-Protocol. Les séquences a exécutées sont désignées par leur numéro et à l'aide de l'instruction PMCR disponible sur les automates CQM1H, C200HX/HG/HE et CS1/CJ1.

Il existe 2 types de protocoles:

- Protocoles Système déjà présents dans les cartes de communications CQM1H et CS1/CJ1
- Protocoles clients élaborés à l'aide de Cx-Protocol puis transférés dans la carte PMSU

De plus, les cartes PMSU disposent de la fonction "Trace" permettant l'enregistrement chronologique des trames échangées.

1.2 Raccordement RS232C



2Configuration

2.1 Automate CS1

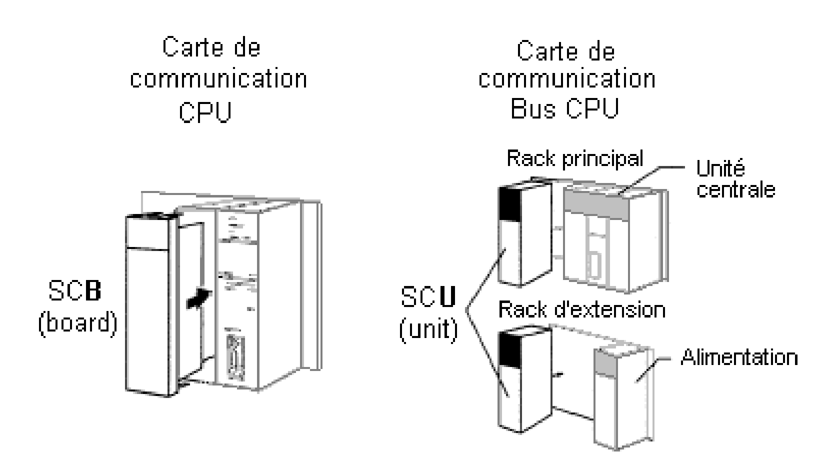
Il existe 2 types de carte PMSU:

Les cartes SCB insérées directement dans l'UC :

- CS1W-SCB21: 2 x RS232C
- CS1W-SCB41: RS232C + RS422A/485

Les cartes SCU montage rack :

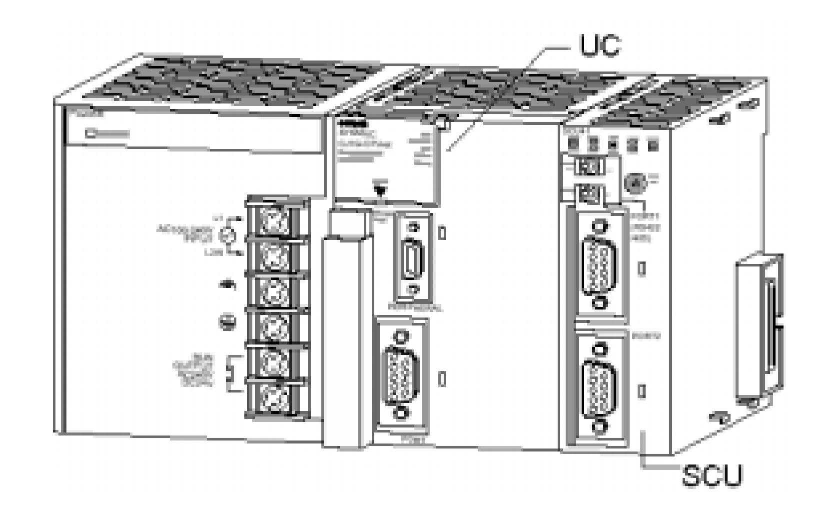
- CS1W-SCU21: 2 x RS232C



2.2 Automate CJ1

Une carte SCU montage rack :

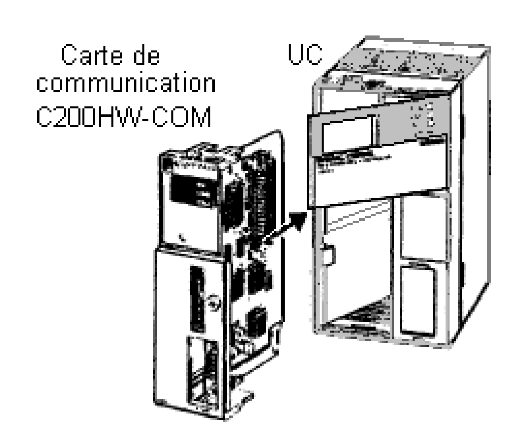
- CJ1W-SCU41: RS232C + RS422A/485



2.3 Automate C200HX/HG/HE

La série alpha dispose de 3 cartes de communication à insérer directement dans l'UC:

- C200HW-COM04-EV1: 1 x RS232C
- C200HW-COM05-EV1: 2 x RS232C
- C200HW-COM06-EV1: RS232C + RS422A/485



Les cartes non EV1 ne disposent pas de toutes les fonctionnalités décrites dans ce manuel.

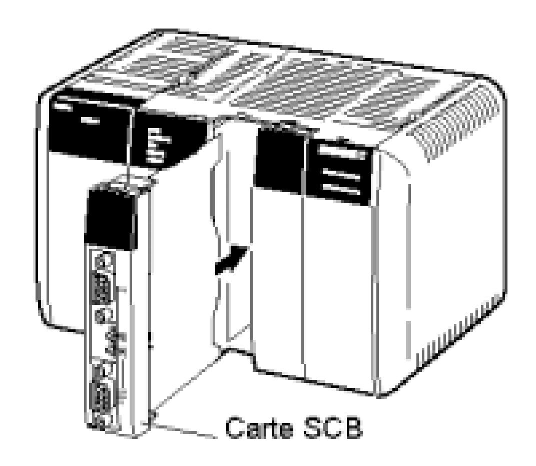
2.4 Automate CQM1H

Il existe une carte SCB insérée directement dans l'UC :

- CQM1H-SCB41 : RS232C + RS422A/485

Pour utiliser le Protocol Macro sur la série CQM1H, il est impératif de:

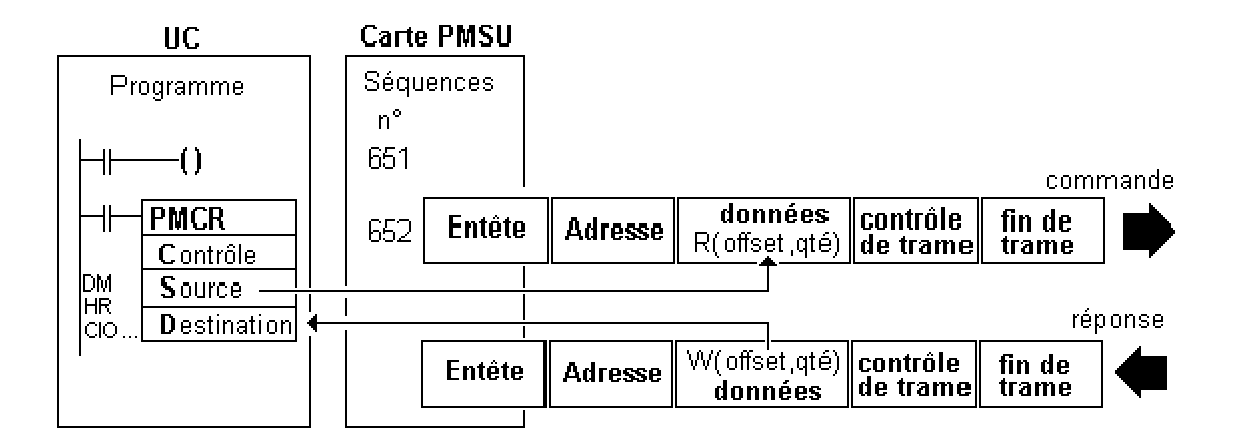
- Basculer le dip switch 8 en façade sur la position on.
- De sélectionner, dans Cx-Protocol, un API de type C200HG-CPU43



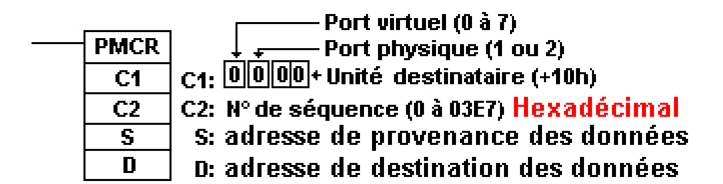
3Principe de Protocol Macro

Le protocole de communication est élaboré à l'aide de Cx-Protocol sous forme de trames appelées messages à transmettre ou bien à recevoir. Ces messages peuvent être enchaînés dans une séquence ou bien suivant une matrice en fonction du contenu des réponses. Le protocole est compilé puis transféré dans la carte PMSU. L'instruction PMCR exécute une séquence en la désignant par son numéro dans le bloc de contrôle C. Les données à envoyer sont lues dans le bloc source (S) tandis que les réponses sont consignées dans le bloc de destination (D).

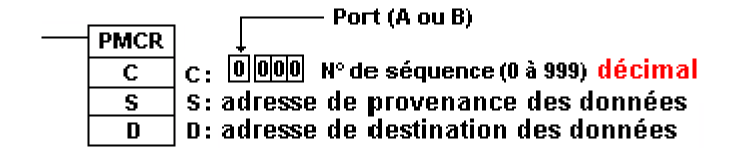
Les données lues [R] ou écrites [W] sont consignées aux adresses S et D décalées d'un offset déterminé dans la séquence.



3.1 L'instruction PMCR sur CS1/CJ1



3.2 L'instruction PMCR sur C200HX/HG/HE et CQM1H

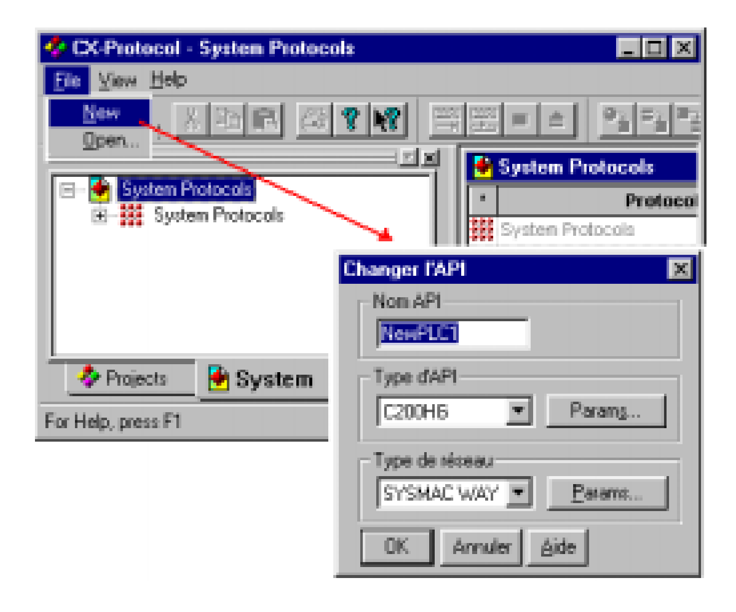


4 Elaboration d’un Protocole

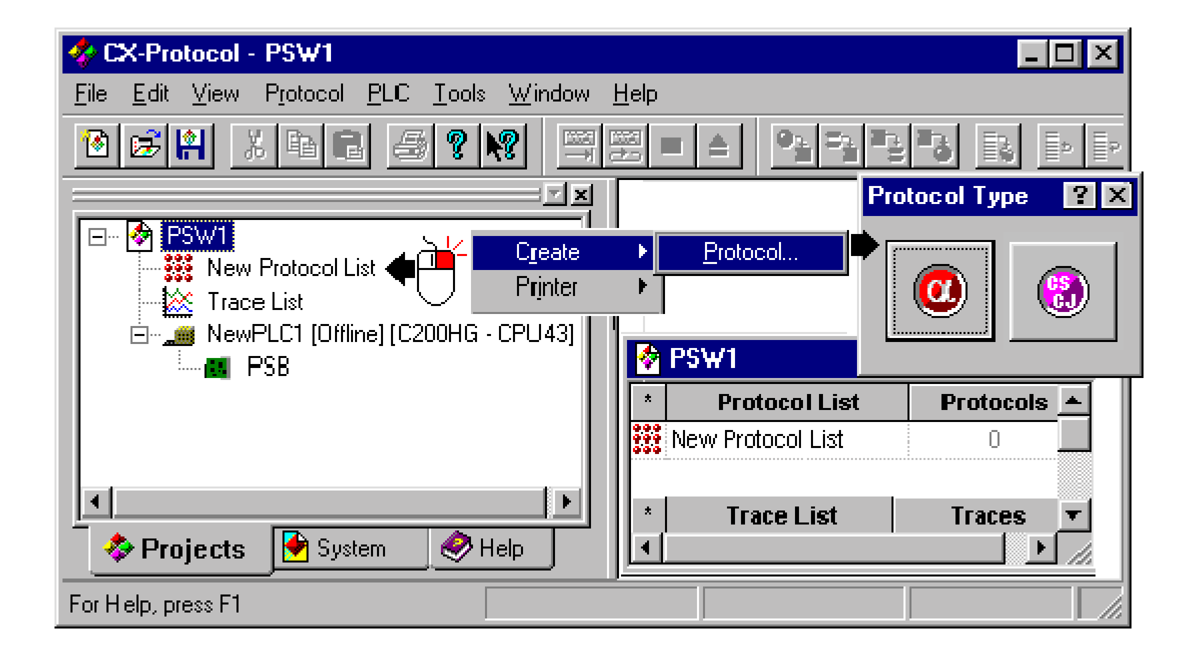
Cx-Protocol dispose de protocoles standard Omron dans l'onglet [System]. Ces protocoles peuvent être copiés dans l’espace [Projet], puis modifiés éventuellement et transférés ensuite dans la carte PMSU.

4.1 Création d'un protocole

Le type d'automate est déclaré au moment de la création du protocole.

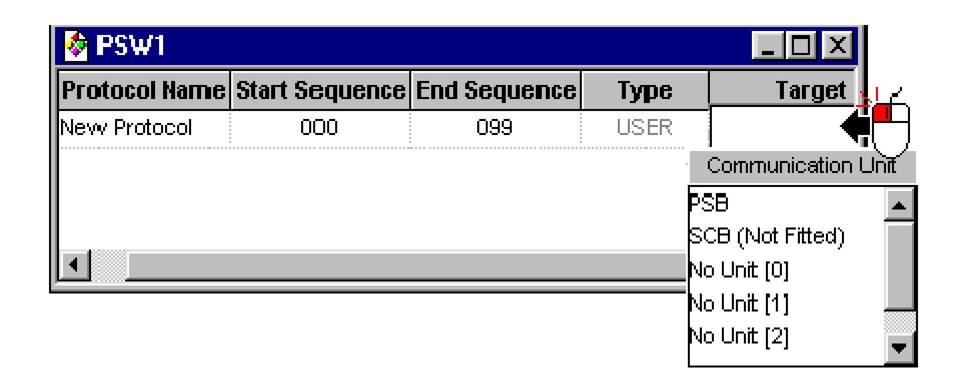


Le menu contextuel de [New Protocol List] permet d'ajouter un protocole destiné à un API de la série Alpha (C200HX…) ou bien CS1/CJ1.



4.2 Sélection de la carte de communication PMSU

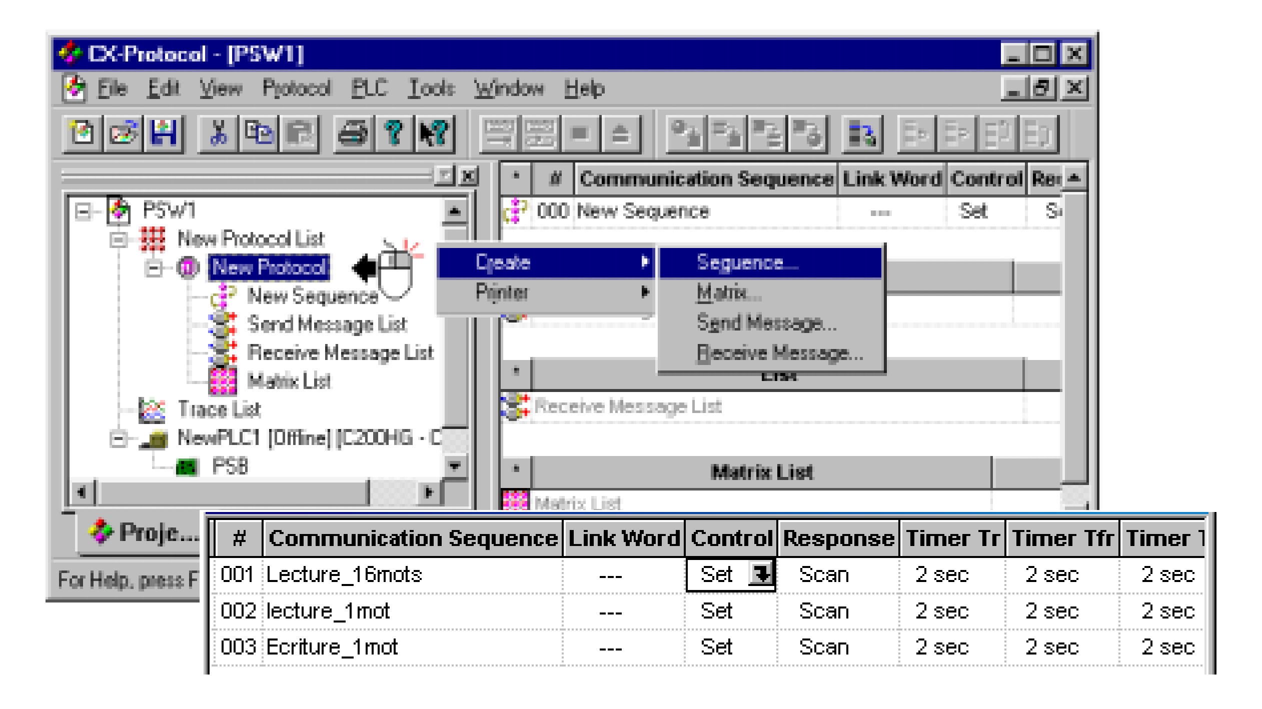
Sélectionnez dans l'éditeur de protocole le type de carte PMSU.



4.3 Elaboration des séquences

4.3.1 Création d'une séquence

Le menu contextuel de [New Protocol] permet la création des séquences.



*	#	Communication sequence	Link word	Control	Response	Timer Tr	TimerTfr	Timer Tfs
	000	Séquence lecture	---	Set	Scan	0.3s		
	001	Séquence écriture	---					

Le numéro et le nom des séquences sont modifiables directement dans l'éditeur.

Link word Permet de déterminer un zone mémoire partagée entre l'UC et la carte PMSU commune à toutes les séquences O1, I1, O2 et I2.

Control Positionne les signaux de contrôle de la transmission RTS/CTS, Xon /Xoff, délimiteurs, etc…(voir plus bas).

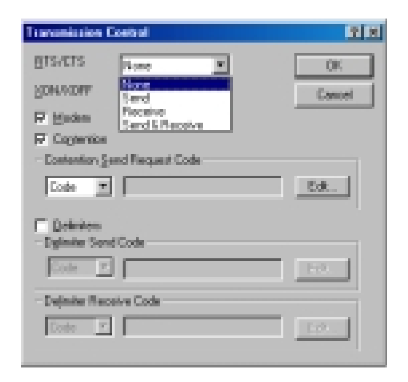
Response **[**Scan] consigne la réponse en fin de cycle.
[Interrupt fixed] consigne la réponse aussitôt et exécute le sous programme (alpha) ou tâche d'interruption (CS1/CJ1) spécifiée.
[Interrupt receive case] consigne la réponse aussitôt et exécute le sous programme (alpha) ou la tâche interruptive (CS1/CJ1) spécifiée pas l'étape dans la séquence.

Timer Tr Délai maxi de réponse. Au delà, une erreur d'exécution est générée.

Timer Tfr Délai maxi entre le premier caractère et le dernier caractère reçu.

Timer Tfs Délai maxi entre le premier caractère et le dernier caractère envoyé.

4.3.2 Méthode de contrôle des signaux de transmissions



RTS/CTS : permet de gérer le flux de données en positionnant le signal CTS de ON à OFF pour informer l'émetteur que son buffer est plein et qu'il ne peut plus recevoir de données.

Xon/Xoff : permet de gérer le flux en envoyant (avec les données) le caractère Xoff (13H) pour informer l'émetteur que son buffer est plein et qu'il ne peut plus recevoir de données.

Modem : fonction spécifique positionnant le signal DTR à ON pendant toute la durée d'exécution de la séquence initiée par PMCR. Le signal RTS est positionné à ON seulement pendant l'émission de caractères.

Contention :permet l'envoi d'un caractère spécifique lorsque le protocole nécessite une demande préalable à émettre.

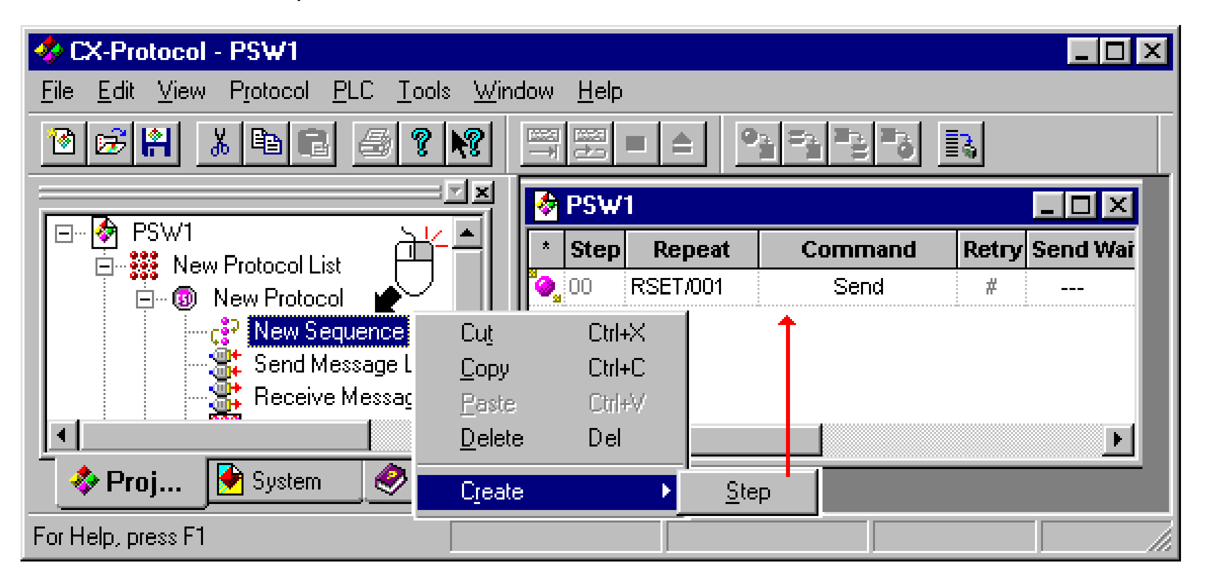
Delimiter : La transmission de plusieurs trames ne peut se poursuivre qu'à condition d'avoir reçu un caractère confirmant la réception (receive code). Dans ce cas la trame émise comporte également un délimiteur de fin de trame (send code).

4.3.3 Création d'une étape

Une séquence peut contenir jusqu'à 16 étapes. Une étape consiste en une commande de:

- Lecture (Receive)
- Ecriture (Send)
- Lecture /écriture (Send /Receive)
- RAZ du tampon de réception (Flush) *
- Attente (Wait) *
- Activation /désactivation du signal DTR (Open/Close) *

*: CS1/CJ1 uniquement



*	Step	Repeat	Command	Retry	Send Wait	Send Message	Recv Message	Response	Next	Error
	00	RSET/001	SEND	0	0.01s	Lecture1DM	######	Yes	End	Abort

Step n° de l'étape

Repeat nombre de répétition de cette étape

Command envoi, réception ou envoi + réception enchaîné suivant le tableau de réponse prédéfini dans la matrice.

Retry nombre d'essai dans la limite de l'intervalle de temps fixé par les timers (commande Send/ Receive uniquement).

Send Wait délai d'attente avant l'exécution de l'envoi

Send Message désigne un message dans la liste [Send Message List] (cf chap. 1.8).

Recv Message désigne un message dans la liste [Receive Message List] (cf chap. 1.8).

Response Réponse attendue

Next détermine l'action suivante, étape suivante, étape spécifique, fin ou annulation

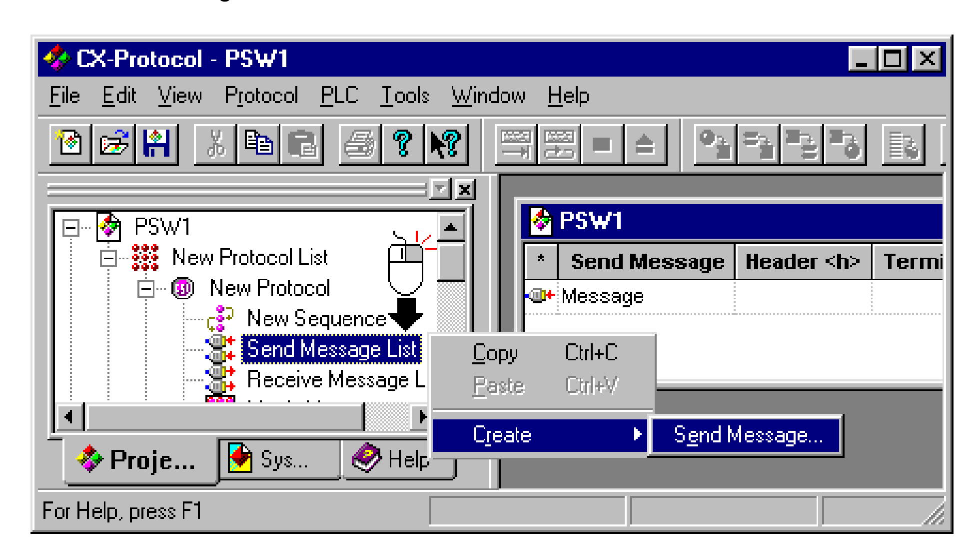
Error détermine l'action suivante en cas d'anomalie (idem next)

4.4 Elaboration des messages

4.4.1 Création d'un message

Les étapes constituant les séquences font appel à des messages décrit dans les listes:

- Send Message List
- Receive Message List



Chacun des champs détermine le type de caractère ou la méthode d'extraction de la donnée et est ensuite concaténé pour former une trame dans le champs [Data].

*	Send Message	Header <h>	Terminator <t>	Check code <c>	Length <l>	Address <a>	Data
	Lecture1DM	"@"	[2A0D]	LRC		$(R(1),2)	<h>+<a>

Send Message Nom du message

Header Caractère d'entête du message

Terminator Caractère de fin du message

Check code méthode de contrôle du message (LRC, LRC2, CRC-CCIT, CRC16, SUM)

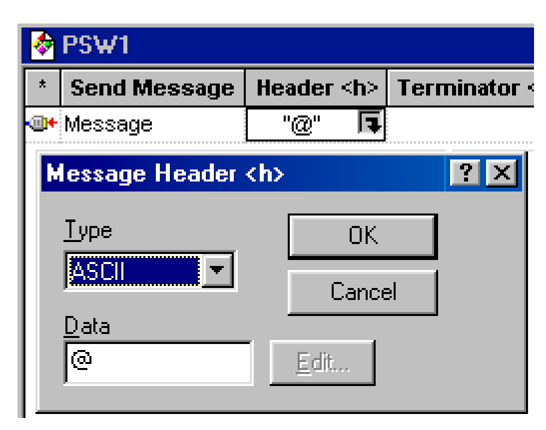
Length méthode de calcul de la taille du message (octet binaire, mots …)

Address détermine à quel endroit se trouve l'adresse dans le bloc de mot pointé par l'instruction PMCR et le nombre d'octet à considérer.

Data concaténation du message.

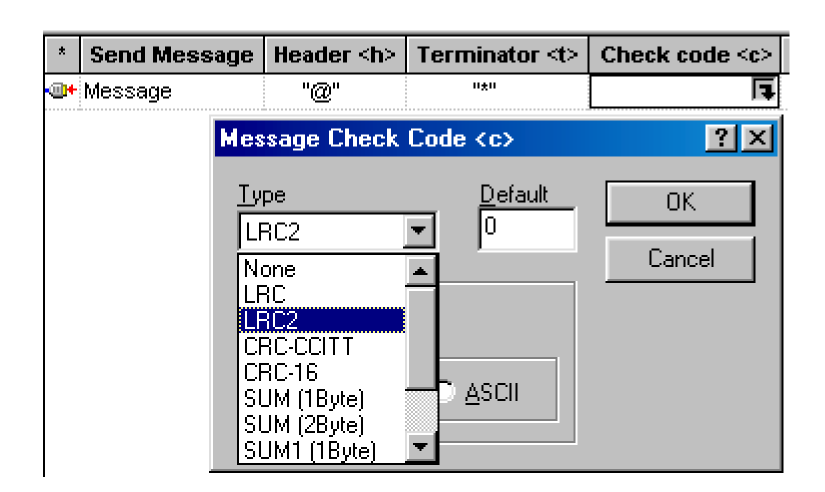
4.4.2 Détermination des délimiteurs de trame

Les champs [Header] et [Terminator] permettent de déterminer un caractère d'en-tête et de fin de trame. Il peut s'agir d'un caractère ou bien du code ASCII d'un caractère non imprimable s'il s'agit d'un protocole binaire.



4.4.3 Détermination de la méthode de contrôle d'erreur

Le champs [Check code] établit le type de contrôle d'erreur, LRC, CRC….
Le code de contrôle d'erreur est calculé puis placé dans la trame à expédier.
Si, lors de la réception, le code reçu est différent du code calculé, une erreur est générée et peut être exploitée ensuite dans une matrice.



4.4.4 Détermination du champs longueur [Length]

Certain protocole (Modbus) introduise dans la trame la longueur de la trame. Si ce champs est placé dans la trame, la longueur de la trame sera calculée automatiquement.
Attention, la longueur de la trame n'est pas vérifiée dans le cas d'un API de la série alpha. Dans le cas d'un CS1/CJ1, une erreur est générée.

4.4.5 Détermination de l'adresse du destinataire [Address]

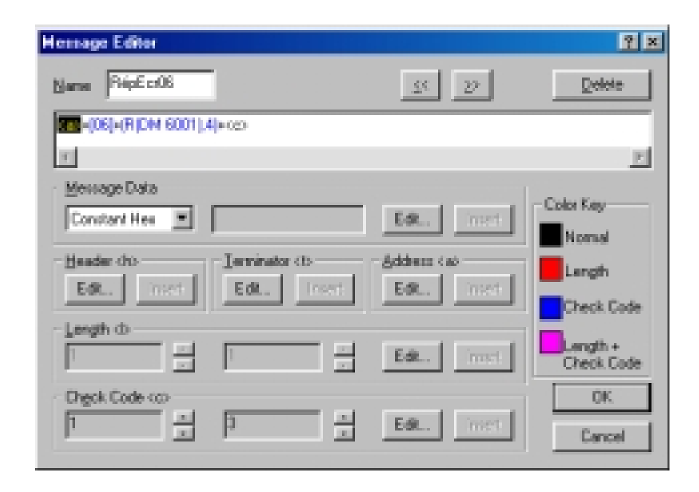
S'il s'agit d'un protocole maître susceptible d'interroger un esclave parmi d'autres, le champs [Address] permet de déterminer l'endroit ou est consigné le numéro de l'esclave. Ce numéro peut être constant ou bien consigné à une adresse relative (cf ch. 4.4.7 Adressage relatif). Dans ce cas, Il peut être judicieux d'utiliser la même adresse que celle utilisée par le compteur de répétition en guise de numéro d'esclave de manière à interroger n esclaves à l'aide d'une seule étape.



4.4.6 Détermination de la trame [Data]

Le champs [Data] permet d'assembler les différents termes qui vont constituer la trame finale. On peut ainsi combiné :

- Un en-tête [Header]
- Une adresse d'esclave [Address]
- Une donnée [Message Data] du type :
  – constante
  – variable déterminé par un adressage relatif (cf. ch 4.4.7 adressage relatif)
- Calcul de longueur de trame [Length]
- Code de contrôle d'erreur [Check code]
- Caractère de fin de trame [Terminator]



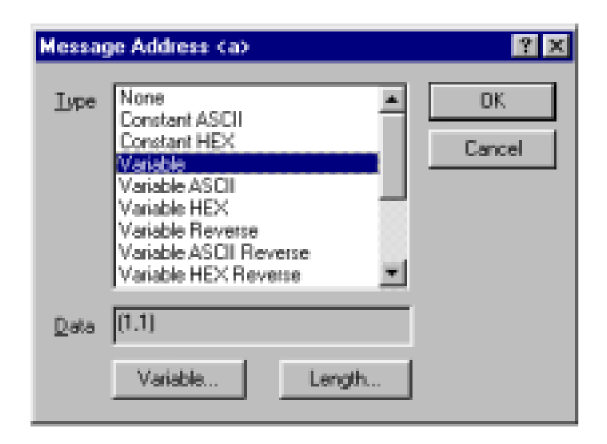
Les couleurs signalent les champs intégrés aux calculs de longueur, de contrôle d'erreur ou les deux en même temps.

4.4.7 L'adressage relatif

4.4.7.1 Principe

Les données utilisées par les champs [Length] et [Data] peuvent être déterminée par une constante ou bien provenir d'une variable mémoire de l'automate dont l'adresse et la taille sont spécifiés à l'aide de :

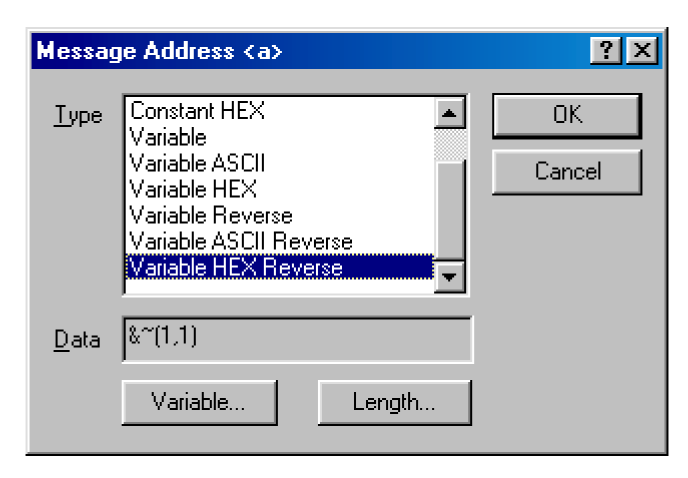
- L'adresse de référence de la variable [Variable…]
- La taille de la variable [Length…]



Ces deux paramètres peuvent être déterminés de 3 façons :

1. Par l'opérande de l'instruction PMCR, [S] pour un envoi et [D] pour une réception
2. Par les mots de la zone commune aux séquences « Link word »
3. Directement (zones CIO, WR, DM, LR, HR, AR, EM)

Les variables ASCII (conversion Hexa->ASCII) sont précédées du symbole **$** tandis que les variables Hexa (conversion ASCII->Hexa) du symbole **&**. Les " variables reverses" sont précédées du symbole **~** pour indiquer que les données sont lues/ écrites en commençant le mot de poids fort.



4.4.7.2 Influence du compteur de répétition

Si le compteur de répétition a été configuré dans l'éditeur de séquence, sa valeur aura une influence directe sur l'adresse effective de la variable automate.



Dans tous les cas, l'adresse effective résulte de l'adresse de référence additionnée du résultat de l'expression :

$$YN + x$$

Ou :

- *N* représente le compteur de répétitions (défini dans l'éditeur de séquences).
- *y* représente le coefficient multiplicateur (pente)
- *x* le décalage

Par défaut, N = 1 , x = 1 et y = 0 ce qui a pour conséquence de ne pas modifier l'adresse de référence : *adresse effective = adresse de référence + 1*

Ce décalage initiale de 1 mot permet de débuter directement à partir du 2ème mot.
En effet, le premier mot contient la taille du bloc pour permettre l'exécution de l'instruction PMCR.

4.4.7.3 Description des champs

Type : Read R() la donnée est lue dans la zone spécifiée par **Data**
Write W() les données sont consignées dans la zone spécifiée par **Data**

yN + x: champs destinés à y et x respectivement

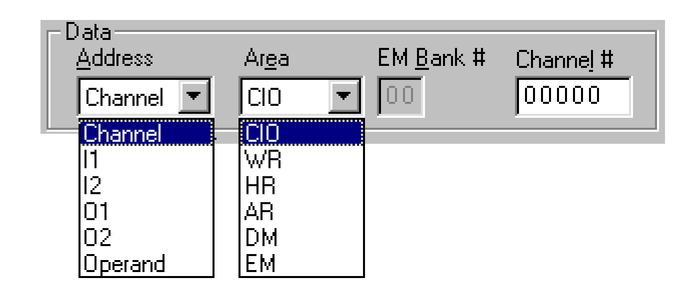
* variable non connue au préalable (en réception)



Data : Channel zone mémoire et adresse spécifiées par Area ,EM et Channel #

Operand zone pointée par l'opérande l'instruction PMCR

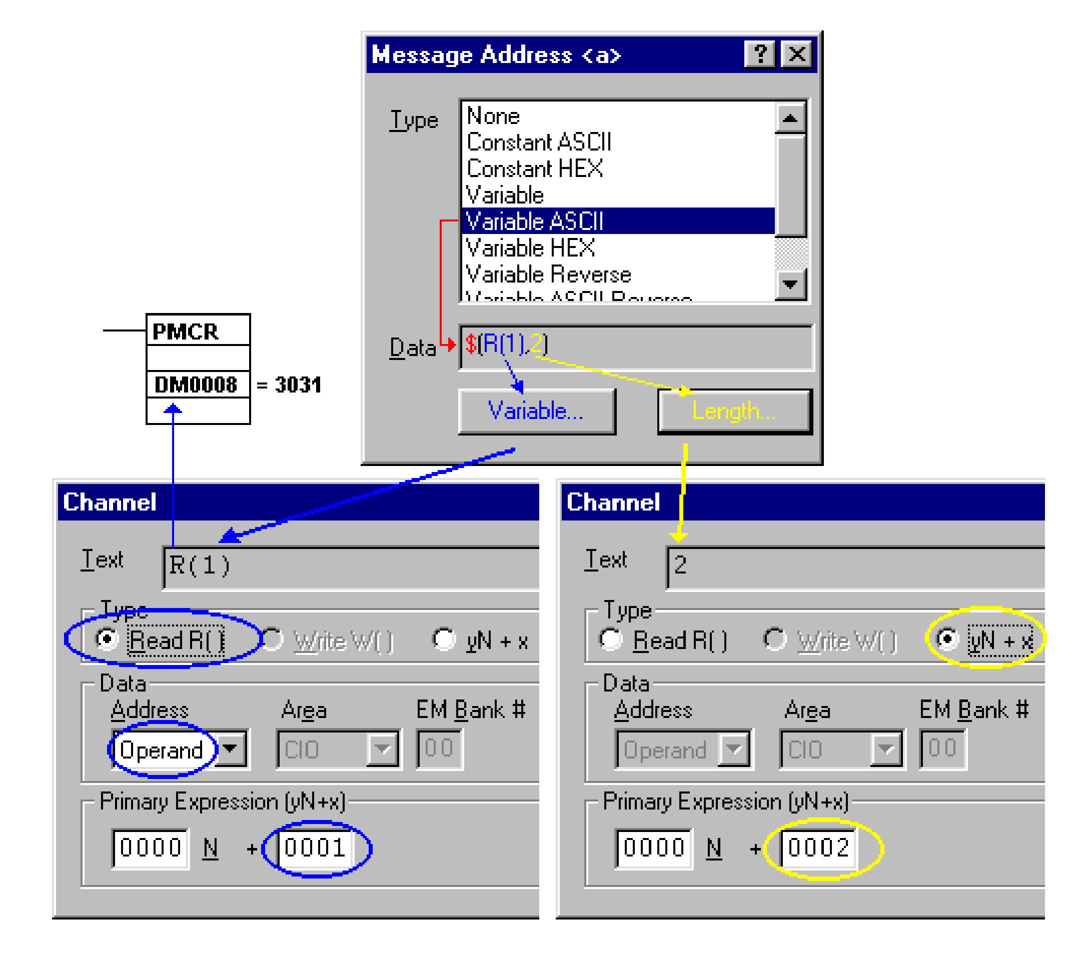
I1,I2,O1,O2 la donnée est lue/écrite dans l'une des 4 zones communes aux séquences (Link area).



4.5 Exemple 1

Envoi d'une trame dont l'adresse à lire est consignée dans le DM0008 (3031).
Cette donnée est toujours exprimée sur 2 octets.

La variable s'écrit "$(R(1),2)" :
$: indique que la valeur doit être convertit en caractère ASCII.
R: indique que la donnée est pointée par l'opérande source de l'instruction PMCR.
(1): le premier mot constitue l'adresse de départ.
2: 2 octets. Le compteur de répétition n'est pas activé "----" par conséquent N est figé à 1.



4.6 Exemple 2

Exécution d'une trame de lecture à 2 adresses différentes 0103 et 0120.

L'étape est donc exécutée 2 fois (compteur de répétition = 2)



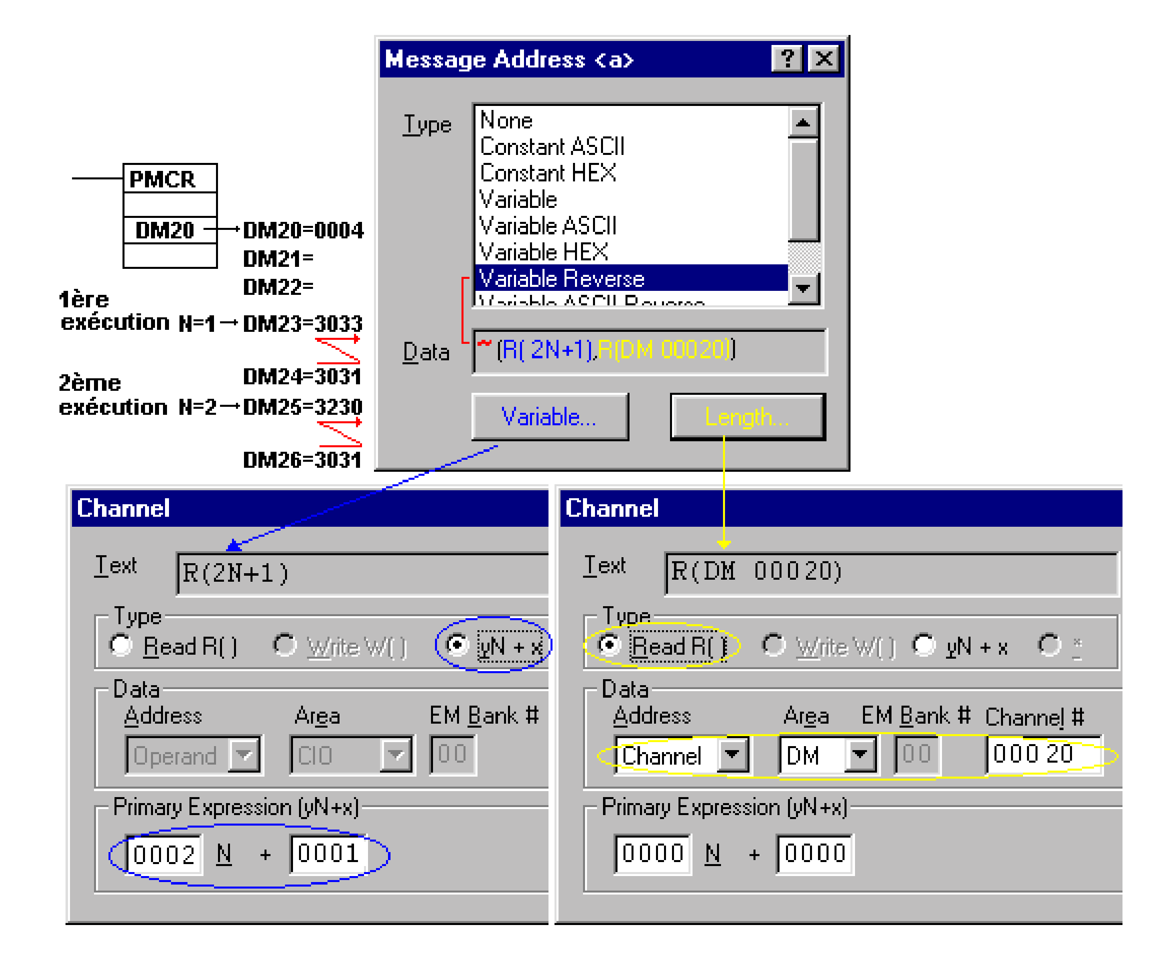
*	Step	Repeat	Command	Retry	Send Wait
	00	RSET/002	Send	#	0.03 sec

La taille de l'adresse peut varier à chaque exécution de PMCR.
L'opérande source de l'instruction PMCR pointe le DM20.
Le DM20 contient la quantité d'octet (4).
Les DM24 et 23 contiennent la première donnée (inversé).
Les DM26 et 25 contiennent la seconde donnée (inversé).
La variable s'écrit: ~(R(2N+1),R(DM 00020))

~: lecture des mots en ordre inverse

R(2N+1): L'adresse est pointée par l'opérande source de l'instruction PMCR avec un décalage de +3 puis de +5 au passage suivant car N vaut successivement 1 puis 2.

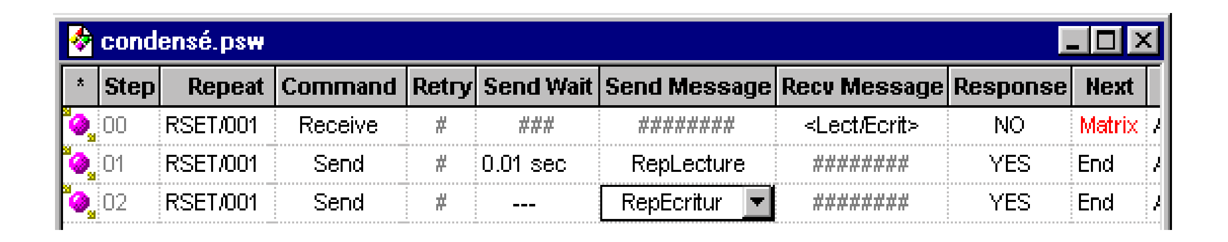
R(DM00020): La quantité d'octet est consignée dans le DM20



4.7 Matrice

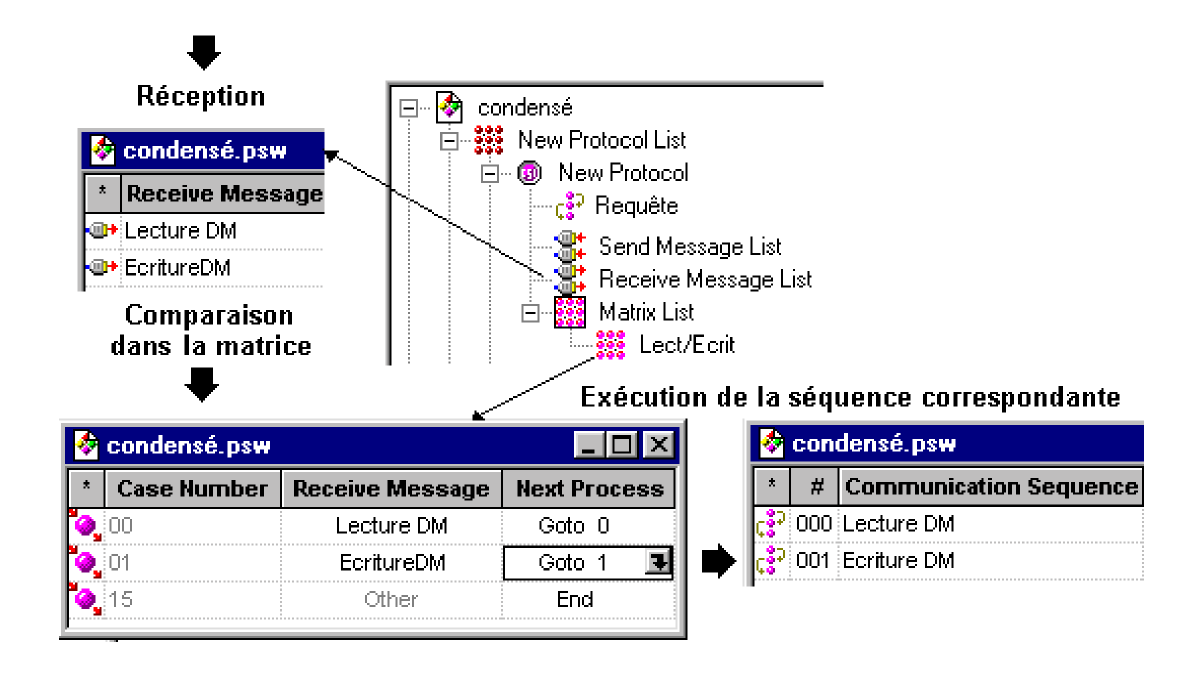
4.7.1 Principe

La séquence renvoi dès le premier pas vers la matrice appelée ici "LectEcrit" qui en fonction de la requête reçu orientera vers le pas spécifié (001 réponse à une lecture ou 002 réponse à une écriture).



condensé.psw

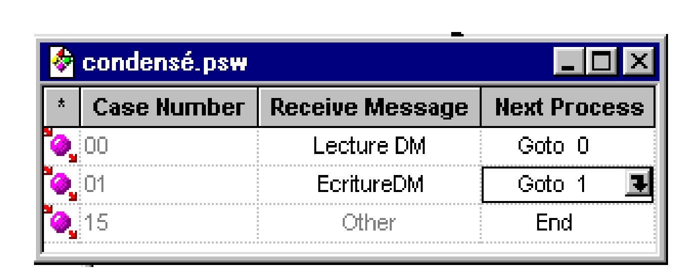
*	Step	Repeat	Command	Retry	Send Wait	Send Message	Recv Message	Response	Next
	00	RSET/001	Receive	#	###	########	<Lect/Ecrit>	NO	Matrix
	01	RSET/001	Send	#	0.01 sec	RepLecture	########	YES	End
	02	RSET/001	Send	#	---	RepEcritur	########	YES	End

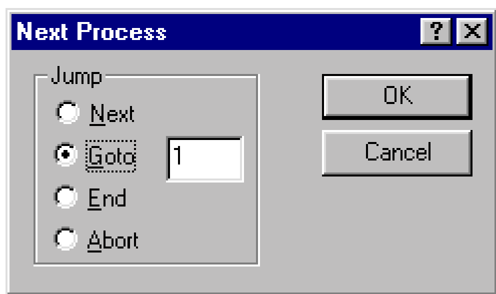


4.7.2 Création d'une matrice

L'option "Matrix" n'apparaît dans le champs Next que lorsqu'une matrice existante est sélectionnée dans le champs Recv Message. Il faut donc dans l'ordre:

1. Créer une matrice en lui donnant simplement un nom.
2. Elaborer la séquence en commençant par le renvoi vers cette matrice puis les messages à transmettre.
3. Elaborer la matrice en désignant les pas à exécuter en fonction des messages reçus et consignés dans Receive Message List





Chaque cas répertorié dans la matrice doit ensuite désigner l'action suivante à entreprendre.

- Next : séquence suivante (dans l'ordre des numéro)
- Goto: saut vers une séquence spécifiée
- End: met fin à l'exécution de la matrice
- Abort: aucune action, la matrice continue de s'exécuter.

Pour être opérationnel, les messages reçus destinés à être comparés dans la matrice, doivent comporter la même entête.

5Mise en œuvre du protocole

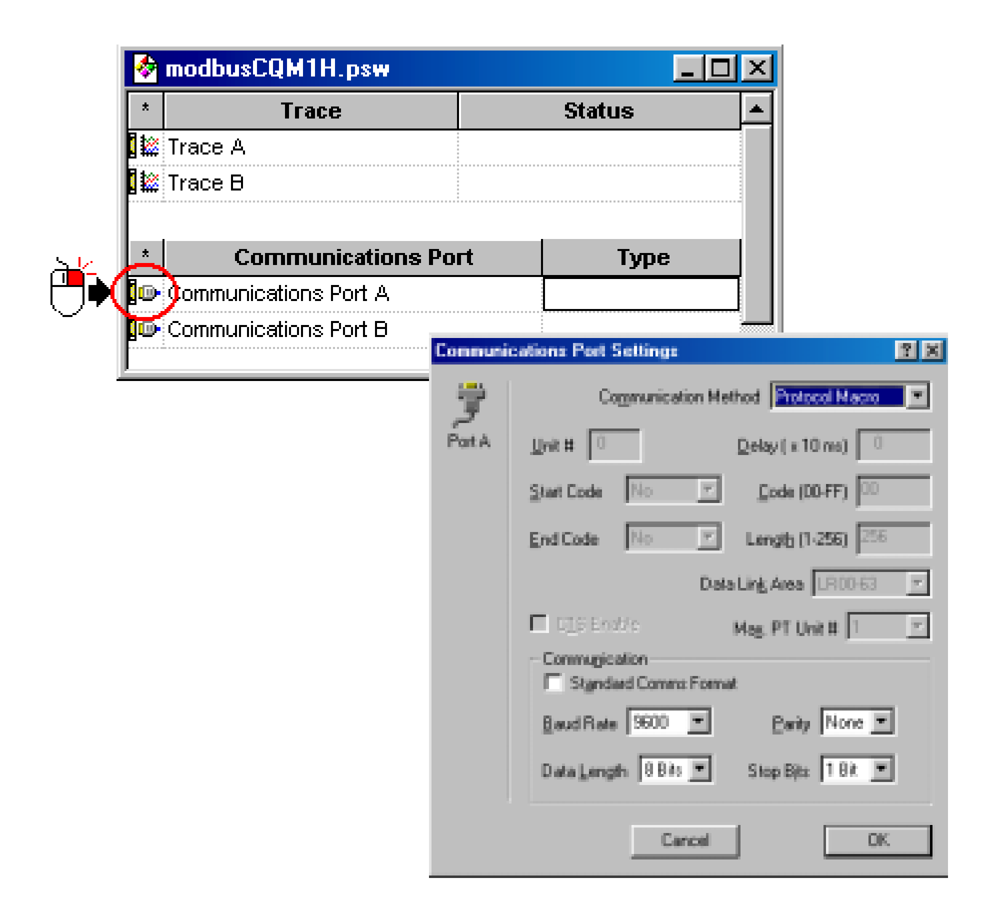
5.1 Configuration de la carte PMSU

La méthode et le format de communication utilisés doivent être configurés dans l'API.

Il faut tout d'abord connecter Cx-Protocole à l'API en choisissant dans la barre des menus l'option [PLC/Connect to PLC] ou bien en passant par le menu contextuel de l'icône



Puis sélectionner par un click droit le port de communication à configurer.
Attention, le menu contextuel n'est accessible qu'en cliquant très précisément sur l'icône 



5.2 Transfert du protocole macro

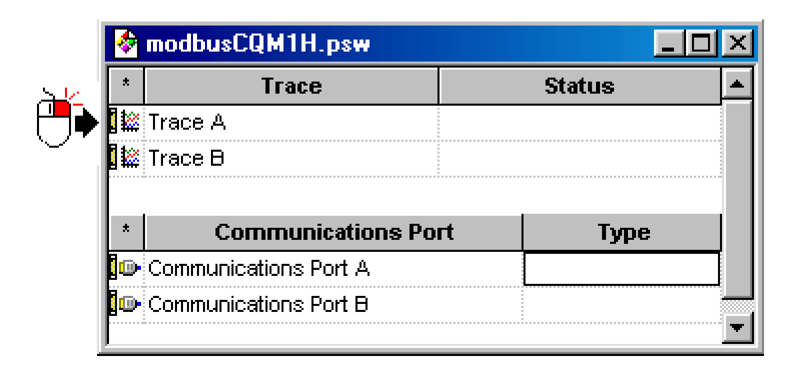
L'option [dowload] du menu [Protocol] (ou bien du menu contextuel) n'est accessible qu'à condition d'avoir sélectionné dans l'arborescence un protocole ou bien la liste contenant les protocoles. Ceci permet de transférer tout ou partie du protocole.

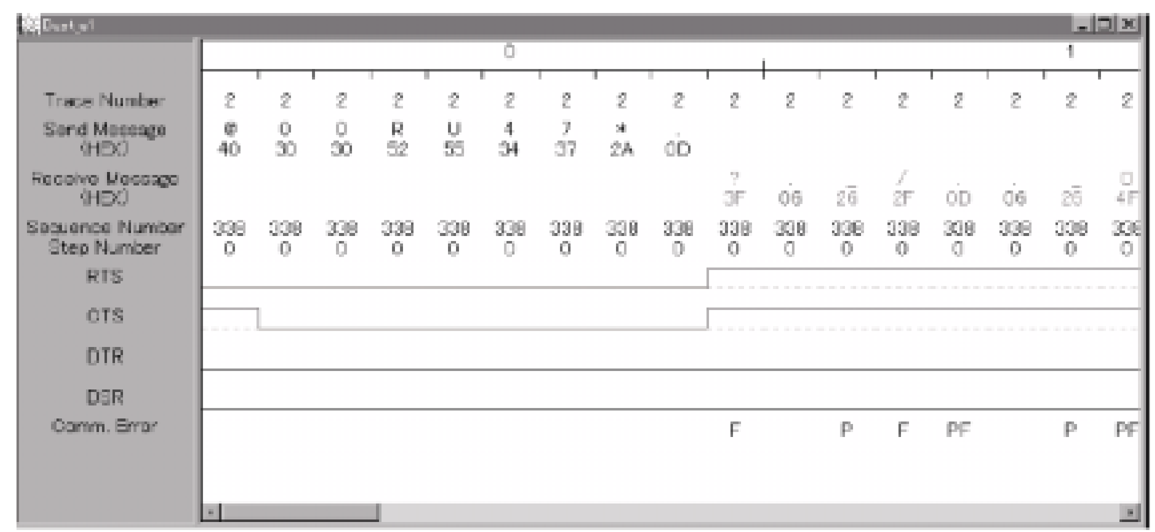
Cx-protocol compile puis transfert le protocole mais propose aussi d'y inclure ou non les sources ainsi qu'une protection par mot de passe.

5.3 La fonction Trace

Cette fonctionnalité de Cx-protocol est très utile pour réaliser la mise au point du protocole. La carte PMSU est capable de tracer l'envoi/réception de 670 caractères sur série alpha et 1700 sur CS1/CJ1.

La fonction trace est démarrée/arrêtée puis éditer par le menu contextuel de l'icône 





5.4 Drapeaux système

Fonctions	CS1-SCB		CS1/CJ1-SCU		HX/HG/HE-SCB	
	Port 1	Port 2	Port 1	Port 2	Port A	Port B
	CIO	CIO	CIO	CIO	IR	IR
Port activé	1906.00	1916.00	n+6.00	n+16.00	---	---
Erreur d'exécution du P. macro	1909.15	1919.15	n+9.15	n+19.15	289.08	289.12
Erreur sur une étape de séquence	1909.14	1919.14	n+9.14	n+19.14	289.09	289.13
Drapeau d'annulation	1909.13	1919.13	n+9.13	n+19.13	---	---
Drapeau fonction trace en cours	1909.12	1919.12	n+9.12	n+19.12	---	---
Drapeau séquence terminée	1909.11	1919.11	n+9.11	n+19.11	289.10	289.14
Drapeau séquence annulée	1909.10	1919.10	n+9.10	n+19.10	289.11	289.15
Drapeau étape d'attente	1909.09	1919.09	n+9.09	n+19.09	---	---
Code erreur : 1 : pas de protocole (alpha) 2 : erreur de n° de séquence 3 : erreur zone L/E données 4 : erreur de syntaxe P. macro 5 : erreur UC	1909.00 à 1909.03	1919.00 à 1919.03	n+9.00 à n+9.03	n+19.00 à n+19.03	286.08 à 286.11	289.12 à 289.15
n° de la séquence en cours	1910	1920	n+10	n+20	---	---
Etape et cas identifié par la matrice	1911	1921	n+11	n+21	287	288
Drapeau mémoire du cas identifié	1912	1922	n+12	n+22	---	---
Drapeau mémoire de l'étape	1913	1923	n+13	n+23	---	---
Valeur courante du compt. de rép.	1914	1924	n+14	n+24	---	---
RAZ port de communication	A636.01	A636.02	A62n.01	A62n.02	IR289.00	IR289.01
Délai réponse PMSU dépassé	A424.00		A417.n		IR268.00	
Erreur bus carte insérée (inner)	A424.01		---		IR268.01	
Erreur de données (CRC, FCS)	A424.09		---		IR268.02	